Fault Tolerant Magnetoresistive Solid-state Storage Device

The present invention relates in general to a magnetoresistive solid-state storage device and to a method for controlling a magnetoresistive solid-state storage device. In particular, but not exclusively, the invention relates to a magnetoresistive solid-state storage device employing error correction coding.

A typical solid-state storage device comprises one or more arrays of storage cells for storing data. Existing semiconductor technologies provide volatile solid-state storage devices suitable for relatively short term storage of data, such as dynamic random access memory (DRAM), or devices for relatively longer term storage of data such as static random access memory (SRAM) or non-volatile flash and EEPROM devices. However, many other technologies are known or are being developed.

Recently, a magnetoresistive storage device has been developed as a new type of non-volatile solid-state storage device (see, for example, EP-A-0918334 Hewlett-Packard). The magnetoresistive solid-state storage device is also known as magnetic random access memory (MRAM) device. MRAM devices have relatively low power consumption and relatively fast access times, particularly for data write operations, which renders MRAM devices ideally suitable for both short term and long term storage applications.

A problem arises in that MRAM devices are subject to physical failure, which can result in an unacceptable loss

of stored data. Currently available manufacturing techniques for MRAM devices are subject to limitations and as a result manufacturing yields of commercially acceptable MRAM devices are relatively low. Although better manufacturing techniques are being developed, these tend to increase manufacturing complexity and cost. Hence, it is desired to apply lower cost manufacturing techniques whilst increasing device yield. Further, it is desired to increase cell density formed on a substrate such as silicon, but as the density increases manufacturing tolerances become increasingly difficult to control, again leading to higher failure rates and lower device yields. Since the MRAM devices are at a relatively early stage in development, it is desired to allow large scale manufacturing of commercially acceptable devices, whilst tolerating the limitations of current manufacturing techniques.

An aim of the present invention is to provide a magnetoresistive solid-state storage device which is tolerant of at least some failures. Another aim is to provide a method for controlling a magnetoresistive solid-state storage device to tolerate at least some failures.

A preferred aim is to provide a magnetoresistive solid-state storage device and a method for controlling such a device which is tolerant of both systematic and random failures. Other preferred aims are to provide a magnetoresistive solid-state storage device and a method for controlling such a device, which allows at least some failures to be tolerated without any loss of stored data, preferably which is efficient to implement, preferably which allows lower cost manufacturing techniques to be

employed, and preferably which allows device yield to be increased.

According to a first aspect of the present invention there is provided a method for controlling a magnetoresistive solid-state storage device having a plurality of storage cells for storing a block of ECC encoded data, the method comprising the steps of: accessing a set of the plurality of storage cells; and determining whether information is unrecoverable from a block of ECC encoded data stored in the accessed storage cells.

In a first preferred embodiment, determination of whether information is unrecoverable from the stored block of ECC encoded data is made by attempting to perform ECC decoding. If the ECC decoding successfully recovers information from the block of ECC encoded data, then use of that set of storage cells can continue in future read and write access cycles. However, if the ECC decoding fails to recover information from the block of ECC encoded data, then preferably remedial action is taken concerning the set of storage cells. For example, the remedial action involves discarding that set of storage cells such that the set is not available in future read and write cycles.

Optionally, the method comprises identifying failed symbols in the block of ECC encoded data, as an output from the ECC decoding step, and comparing the identified number of failed symbols against a threshold value. The threshold value suitably represents a safety margin, such as 50% to 95% of the maximum number of failed symbols

which can be corrected by ECC decoding the block of ECC encoded data. The safety margin represents the situation where, although a relatively high proportion of failed symbols have been identified in the block of ECC encoded data, it is reasonable to continue using that set of storage cells in future. Even though further systematic or random failures might be encountered in a future read operation, it is reasonable to expect that the number of failed symbols will still be correctable by ECC decoding the block of ECC encoded data.

In a second preferred embodiment of the present invention, the accessed set of storage cells is evaluated based on parametric values, prior to attempting ECC decoding of the block of ECC encoded data. Preferably, the method comprises determining whether original information is expected to be unrecoverable from the block of ECC encoded data stored in the accessed set of storage cells. In particular, it is determined whether original information is expected to be unrecoverable because the probability of failing to correctly perform ECC decoding is unacceptably high. Where original information is not expected to be unrecoverable, then use of the set of storage cells may continue. The first and second embodiments are preferably combined, such that a decision to continue use of the set of storage cells, or take remedial action, is made either after performing a parametric based test as in the second embodiment, or after performing ECC decoding as in the first embodiment, or a decision can be made at either stage.

Preferably, in the second embodiment, the method comprises determining, from accessing the set of storage

cells, failed symbols in the block of ECC encoded data that have been affected by a physical failure. Suitably, a determination is made whether there are more failed symbols in the block of ECC encoded data than can be corrected by error correction decoding the block of ECC encoded data. Here, a situation is identified where, due to physical failures, ECC decoding the block of ECC encoded data may well fail to recover the original information. In other words, there is an unacceptable probability that decoding the block of ECC encoded data will not correctly recover original information.

Preferably, accessing the set of storage cells comprises obtaining parametric values, which are compared against one or more ranges. Suitably, for most of the accessed set of storage cells, a logical bit value is derived, but some of the storage cells can be identified as being affected by a physical failure. Suitably, a failure count is determined based on the identified failed cells. The failure count can simply represent the number of failed cells, but preferably the failure count is based on failed symbols of the block of ECC encoded data affected by the identified failed cells. Preferably, the failure count is compared against a threshold value. As one option, the threshold value represents the total number of failed symbols which can be corrected by ECC decoding the block of ECC encoded data. As a second option, the threshold value represents a safety margin less than the total number of failed symbols correctable by ECC decoding, such as between about 50% to 95% of the total number. In this situation the threshold value is particularly useful in that only some types of physical failures in MRAM devices can be readily identified from

the obtained parametric values, and the threshold value is set such that, given the identified number of failures, it is still reasonable to perform ECC decoding, whilst allowing for an additional number of as yet unidentified failures to affect the block of ECC encoded data.

Conveniently, original information is received for storing in the MRAM device in units of a sector, such as 512 bytes. The original information sector is error correction encoded to form one or more blocks of ECC encoded data. In the preferred embodiment a linear ECC scheme such as a Reed-Solomon code is employed. Conveniently, each sector of original information is encoded to form a sector of ECC encoded data comprising four codewords. Each codeword suitably forms the block of ECC encoded data mentioned above.

According to a second aspect of the present invention there is provided a method for controlling a magnetoresistive solid-state storage device, comprising the steps of: receiving original information which it is desired to store; error correction encoding the original information to form a block of ECC encoded data; storing the block of ECC encoded data in a set of magnetoresistive storage cells arranged in at least one array; accessing the set of storage cells; forming logical symbol values of the block of ECC encoded data from the accessed set of storage cells; error correction decoding the block of ECC encoded data to provide recovered information; if the decoding step provided recovered information then outputting the recovered information and continuing use of the set of storage cells, or else if the decoding step did

not provide recovered information then taking remedial action in respect of the set of storage cells.

Preferably, the method comprises identifying, from the ECC decoding, zero or more failed symbols in the block of ECC encoded data; comparing the identified number of failed symbols against a threshold value; and, if the ECC decoding did not recover original information, or if the identified number of failed symbols is greater than the threshold value, then taking remedial action concerning the accessed set of storage cells.

According to a third aspect of the present invention there is provided a method for controlling a magnetoresistive solid-state storage device, comprising the steps of: receiving original information which it is desired to store; error correction encoding the original information to form a block of ECC encoded data; storing the block of ECC encoded data in a set of magnetoresistive storage cells arranged in at least one array; accessing the set of storage cells; comparing parametric values obtained by accessing the set of storage cells against one or more ranges; identifying failed cells amongst the accessed set of cells; forming a failure count based on the identified failed cells; comparing the failure count against a threshold value; and determining whether the original information is expected to be unrecoverable from the block of ECC encoded data stored in the accessed set of storage cells.

According to a fourth aspect of the present invention there is provided a magnetoresistive solid-state storage device, comprising: at least one array of magnetoresistive

storage cells; a ECC encoding unit for forming a block of ECC encoded data from a unit of original information; and a controller arranged to store the block of ECC encoded data in a set of the storage cells, access the set of storage cells, and determine whether the original information is unrecoverable from the block of ECC encoded data stored in the accessed set of storage cells.

For a better understanding of the invention, and to show how embodiments of the same may be carried into effect, reference will now be made, by way of example, to the accompanying diagrammatic drawings in which:

Figure 1 is a schematic diagram showing a preferred MRAM device including an array of storage cells;

Figure 2 shows a preferred logical data structure;

Figure 3 shows an overview of a preferred method for controlling an MRAM device;

Figure 4 shows a first preferred method for controlling an MRAM device;

Figure 5 shows a second preferred method for controlling an MRAM device; and

Figure 6 is a graph illustrating a parametric value obtained from a storage cell of an MRAM device.

To assist a complete understanding of the present invention, an example MRAM device will first be described with reference to Figure 1, including a description of the

failure mechanisms found in MRAM devices. The preferred methods for controlling such MRAM devices will then be described with reference to Figures 2 to 6.

Figure 1 shows a simplified magnetoresistive solid-state storage device 1 comprising an array 10 of storage cells 16. The array 10 is coupled to a controller 20 which, amongst other control elements, includes an ECC coding and decoding unit 22. The controller 20 and the array 10 can be formed on a single substrate, or can be arranged separately.

In one preferred embodiment, the array 10 comprises of the order of 1024 by 1024 storage cells, just a few of which are illustrated. The cells 16 are each formed at an intersection between control lines 12 and 14. In this example control lines 12 are arranged in rows, and control lines 14 are arranged in columns. One row 12 and one or more columns 14 are selected to access the required storage cell or cells 16 (or conversely one column and several rows, depending upon the orientation of the array). Suitably, the row and column lines are coupled to control circuits 18, which include a plurality of read/write control circuits. Depending upon the implementation, one read/write control circuit is provided per column, or read/write control circuits are multiplexed or shared between columns. In this example the control lines 12 and 14 are generally orthogonal, but other more complicated lattice structures are also possible.

In a read operation of the currently preferred MRAM device, a single row line 12 and several column lines 14 (represented by thicker lines in Figure 1) are activated

in the array 10 by the control circuits 18, and a set of data read from those activated cells. This operation is termed a slice. The row in this example is 1024 storage cells long *l* and the accessed storage cells 16 are separated by a minimum reading distance *m*, such as sixty-four cells, to minimise cross-cell interference in the read process. Hence, each slice provides up to $l/m = 1024/64 = 16$ bits from the accessed array.

To provide an MRAM device of a desired storage capacity, preferably a plurality of independently addressable arrays 10 are arranged to form a macro-array. Conveniently, a small plurality of arrays 10 (typically four) are layered to form a stack, and plural stacks are arranged together, such as in a 16 x 16 layout. Preferably, each macro-array has a 16 x 18 x 4 or 16 x 20 x 4 layout (expressed as width x height x stack layers). Optionally, the MRAM device comprises more than one macro-array. In the currently preferred MRAM device only one of the four arrays in each stack can be accessed at any one time. Hence, a slice from a macro-array reads a set of cells from one row of a subset of the plurality of arrays 10, the subset preferably being one array within each stack.

Each storage cell 16 stores one bit of data suitably representing a numerical value and preferably a binary value, i.e. one or zero. Suitably, each storage cell includes two films which assume one of two stable magnetisation orientations, known as parallel and anti-parallel. The magnetisation orientation affects the resistance of the storage cell. When the storage cell 16 is in the anti-parallel state, the resistance is at its

highest, and when the magnetic storage cell is in the parallel state, the resistance is at its lowest. Suitably, the anti-parallel state defines a zero logic state, and the parallel state defines a one logic state, or vice versa. As further background information, EP-A- 0 918 334 (Hewlett-Packard) discloses one example of a magnetoresistive solid-state storage device which is suitable for use in preferred embodiments of the present invention.

Although generally reliable, it has been found that failures can occur which affect the ability of the device to store data reliably in the storage cells 16. Physical failures within a MRAM device can result from many causes including manufacturing imperfections, internal effects such as noise in a read process, environmental effects such as temperature and surrounding electro-magnetic noise, or ageing of the device in use. In general, failures can be classified as either systematic failures or random failures. Systematic failures consistently affect a particular storage cell or a particular group of storage cells. Random failures occur transiently and are not consistently repeatable. Typically, systematic failures arise as a result of manufacturing imperfections and ageing, whilst random failures occur in response to internal effects and to external environmental effects.

Failures are highly undesirable and mean that at least some storage cells in the device cannot be written to or read from reliably. A cell affected by a failure can become unreadable, in which case no logical value can be read from the cell, or can become unreliable, in which case the logical value read from the cell is not

necessarily the same as the value written to the cell (e.g. a "1" is written but a "0" is read). The storage capacity and reliability of the device can be severely affected and in the worst case the entire device becomes unusable.

Failure mechanisms take many forms, and the following examples are amongst those identified:

1. Shorted bits - where the resistance of the storage cell is much lower than expected. Shorted bits tend to affect all storage cells lying in the same row and the same column.

2. Open bits - where the resistance of the storage cell is much higher than expected. Open bit failures can, but do not always, affect all storage cells lying in the same row or column, or both.

3. Half-select bits - where writing to a storage cell in a particular row or column causes another storage cell in the same row or column to change state. A cell which is vulnerable to half select will therefore possibly change state in response to a write access to any storage cell in the same row or column, resulting in unreliable stored data.

4. Single failed bits - where a particular storage cell fails (e.g. is stuck always as a "0"), but does not affect other storage cells and is not affected by activity in other storage cells.

These four example failure mechanisms are each systematic, in that the same storage cell or cells are consistently affected. Where the failure mechanism affects only one cell, this can be termed an isolated failure. Where the failure mechanism affects a group of cells, this can be termed a grouped failure.

Whilst the storage cells of the MRAM device can be used to store data according to any suitable logical layout, data is preferably organised into basic data units (e.g. bytes) which in turn are grouped into larger logical data units (e.g. sectors). A physical failure, and in particular a grouped failure affecting many cells, can affect many bytes and possibly many sectors. It has been found that keeping information about cells, bytes or even sectors affected by physical failures is not efficient, due to the quantity of data involved. That is, attempts to produce a list of all logical data units rendered unusable due to at least one physical failure, tend to generate a quantity of management data which is too large to handle efficiently. Further, depending on how the data is organised on the device, a single physical failure can potentially affect a large number of logical data units, such that avoiding use of all bytes, sectors or other units affected by a failure substantially reduces the storage capacity of the device. For example, a grouped failure such as a shorted bit failure in just one storage cell affects many other storage cells, which lie in the same row or the same column. Thus, a single shorted bit failure can affect 1023 other cells lying in the same row, and 1023 cells lying in the same column - a total of 2027 affected cells. These 2027 affected cells may form part

of many bytes, and many sectors, each of which would be rendered unusable by the single grouped failure.

Some improvements have been made in manufacturing processes and device construction to reduce the number of manufacturing failures and improve device longevity, but this usually involves increased manufacturing costs and complexity, and reduced device yields. Hence, techniques are being developed which respond to failures and avoid future loss of data. One example technique is the use of sparing. A row identified as containing failures is made redundant (spared) and replaced by one of a set of unused additional spare rows, and similarly for columns. However, either a physical replacement is required (i.e. routing connections from the failed row or column to instead reach the spare row or column), or else additional control overhead is required to map logical addresses to physical row and column lines. Only a limited sparing capacity can be provided, since enlarging the device to include spare rows and columns reduces device density for a fixed area of substrate and increases manufacturing complexity. Therefore, where failures are relatively common, sparing is unable to cope leading to possible loss of data. Also, sparing is not useful in handling random failures, and involves additional management overhead to determine deployment of sparing capacity.

The preferred embodiments of the present invention employ error correction coding to provide a magnetoresistive solid-state storage device which is error tolerant, preferably to tolerate and recover from both random failures and systematic failures. Typically, error correction coding involves receiving original information

which it is desired to store and forming encoded data which allows errors to be identified and ideally corrected. The encoded data is stored in the solid-state storage device. At read time, the original information is recovered by error correction decoding the encoded stored data. A wide range of error correction coding (ECC) schemes are available and can be employed alone or in combination. Suitable ECC schemes include both schemes with single-bit symbols (e.g. BCH) and schemes with multiple-bit symbols (e.g. Reed-Solomon).

As general background information concerning error correction coding, reference is made to the following publication: W.W. Peterson and E.J. Weldon, Jr., "Error-Correcting Codes", 2nd edition, 12th printing, 1994, MIT Press, Cambridge MA.

A more specific reference concerning Reed-Solomon codes used in the preferred embodiments of the present invention is: "Reed-Solomon Codes and their Applications", ED. S.B. Wicker and V.K. Bhargava, IEEE Press, New York, 1994.

Figure 2 shows an example logical data structure used in preferred embodiments of the present invention. Original information 200 is received in predetermined units such as a sector comprising 512 bytes. Error correction coding is performed to produce a block of encoded data 202, in this case an encoded sector. The encoded sector 202 comprises a plurality of symbols 206 which can be a single bit (e.g. a BCH code with single-bit symbols) or can comprise multiple bits (e.g. a Reed-Solomon code using multi-bit symbols). In the preferred

Reed-Solomon encoding scheme, each symbol 206 conveniently comprises eight bits. As shown in Figure 2, the encoded sector 202 comprises four codewords 204, each comprising of the order of 144 to 160 symbols. The eight bits corresponding to each symbol are conveniently stored in eight storage cells 16. A physical failure which affects any of these eight storage cells can result in one or more of the bits being unreliable (i.e. the wrong value is read) or unreadable (i.e. no value can be obtained), giving a failed symbol.

Error correction decoding the encoded data 202 allows failed symbols 206 to be identified and corrected. The preferred Reed-Solomon scheme is an example of a linear error correcting code, which mathematically identifies and corrects completely up to a predetermined maximum number of failed symbols 206, depending upon the power of the code. For example, a [160,128,33] Reed-Solomon code having one hundred and sixty 8-bit symbols corresponding to one hundred and twenty-eight original information bytes and a minimum distance of thirty-three symbols can locate and correct up to sixteen failed symbols. Suitably, the ECC scheme employed is selected with a power sufficient to recover original information 200 from the encoded data 202 in substantially all cases. Very rarely, a block of encoded data 202 is encountered which is affected by so many failures that the original information 200 is unrecoverable. Also, very rarely the failures result in a mis-correct, where information recovered from the encoded data 202 is not equivalent to the original information 200. Even though the recovered information does not correspond to the original information, a mis-correct is

not readily determined and means that the original information is unrecoverable.

In the current MRAM devices, grouped failures tend to affect a large group of storage cells, lying in the same row or column. This provides an environment which is unlike prior storage devices. The preferred embodiments of the present invention employ an ECC scheme with multi-bit symbols. Where manufacturing processes and device design change over time, it may become more appropriate to organise storage locations expecting bit-based errors and then apply an ECC scheme using single-bit symbols, and at least some the following embodiments can be applied to single-bit symbols.

Figure 3 shows a simplified overview of a preferred method for controlling the MRAM device 1 of Figure 1.

Step 301 comprises accessing a plurality of the storage cells 16 of the MRAM device. Preferably, the plurality of storage cells correspond to a block of encoded data, such as a codeword 204, or an encoded sector 202. Suitably, a plurality of read operations are performed by accessing the plurality of cells 16 using the row and column control lines 12 and 14. The read operations provide logical bit values which are used to form the symbols 206, and the symbols in turn are built into a complete logical block of data such as the codeword 204. In this example, four codewords 204 together form a complete encoded sector 202, from which the original information sector 200 can be recovered.

Step 302 comprises determining whether original information is unrecoverable from the block of encoded data. That is, the step 302 comprises determining whether decoding the block of encoded data is expected not to be able to produce recovered information, or determining whether attempting to decode the block of encoded data does not produce recovered information. The determining step can be performed by ECC decoding the block of encoded data as a logical evaluation technique, or can be performed using physical evaluation techniques, and preferably a combination of both logical and physical techniques are employed as will be described in more detail below.

Where step 302 determines that ECC decoding has not produced recovered information, or is not expected to produce recovered information, then remedial action is taken in step 304. Otherwise, use of the cells continues in step 303.

The remedial action in step 304 may take any suitable form, to manage future activity in the storage cells 16. As one example, the access of step 301 is immediately repeated, in the hope of avoiding some random errors and this time obtaining symbol values for the encoded data from which the original data can be recovered by ECC decoding. As a second example, the set of storage cells 16 corresponding to a failed codeword 204 or to a complete encoded sector 202 are identified and discarded, in order to avoid possible loss of data in future. In the currently preferred embodiments it is most convenient to use or discard sets of storage cells corresponding to a

sector 202, although greater or lesser granularity can be applied as desired.

Figure 4 shows a more detailed preferred method for controlling the MRAM device, using logical evaluation of the accessed set of storage cells 16 corresponding to a block of encoded data such as a codeword 204 or an encoded sector 202.

Step 401 comprises accessing the set of storage cells 16, equivalent to step 301 above.

Step 402 comprises performing ECC decoding of the block of encoded data obtained by accessing the storage cells in step 401.

Step 403 comprises determining whether the ECC decoding of step 402 was not successful, in the sense that the ECC decoding has not produced recovered information from the block of data. Where ECC decoding is not successful, it is not possible to recover the original data 200 from the accessed storage cells 16, and remedial action can be taken as in step 304.

Optionally, the method includes the step 404 of determining the number of failed symbols identified by the ECC decoding of step 402, and comparing the identified number of failures against a threshold value. A physical failure in any of the accessed set of storage cells can result in a failed symbol. The threshold value selected for the comparison is preferably in the range of between about 50% and 95% of the maximum number of failures that can be corrected by performing the ECC decoding of step

402. The threshold value in step 404 is selected on the basis that although a number of failures have been identified in this particular block of data, it is still reasonable to continue using the selected set of storage cells with the expectation of still being able to successfully perform ECC decoding next time those cells are accessed. The threshold value in step 404 provides a safety margin allowing a further failure or failures to occur in the next access, whilst still allowing a successful ECC decoding to be performed.

In almost all practical cases, the ECC scheme employed is sufficiently powerful to provide recovered information equivalent to the original information sector 200. The original information 200 is output from the MRAM device in step 405.

The method of Figure 4 is conveniently employed whilst the MRAM device is in use. Suitably, the method of Figure 4 is applied whilst the device stores variable user data, allowing dynamic management of data storage in the device. For example, it is possible that the number of systematic errors will increase as the device ages. A small number of sets of storage cells such as sectors 202 will become unreliable and should be removed from active use as a remedial action. However, it is expected that most sectors will continue in use reliably, by employing a suitable ECC scheme.

Additionally or alternatively, the method of Figure 4 is conveniently applied when the MRAM device is first manufactured, or is first installed, or at power up, or at convenient times subsequently such as a periodic check.

Suitably, a sample of test data is applied to a block such as a sector, and the test method of Figure 4 performed to establish the suitability of that sector for future use.

Figure 5 shows a second preferred method for controlling the MRAM device 1. As in Figures 3 and 4, the method is intended for use with a logical block of data such as codeword 204 or an encoded sector 202.

In step 501 the set of storage cells corresponding to the block of data are accessed, preferably in a set of read operations.

Step 502 comprises obtaining a plurality of parametric values associated with the accessed set of storage cells from the access of step 401. Suitably, a read voltage is applied along the row and column control lines 12, 14 causing a sense current to flow through selected storage cells 16, which have a resistance determined by parallel or anti-parallel alignment of the two magnetic films. The resistance of a particular cell is determined according to a phenomenon known as spin tunnelling and the cells are often referred to as magnetic tunnel junction storage cells. The condition of the storage cell is determined by measuring the sense current (proportional to resistance) or a related parameter such as response time to discharge a known capacitance.

Step 503 comprises comparing the obtained parametric values to one or more predicted ranges. The comparison of step 503 in almost all cases allows a logical value (e.g. one or zero) to be established for each cell. However, the comparison also conveniently allows at least some

forms of physical failure to be identified. For example, it has been determined that a shorted bit failure leads to a very low resistance value in all cells of a particular row and a particular column. Also, open-bit failures can cause a very high resistance value for all cells of a particular row and column. By comparing the obtained parametric values against predicted ranges, cells affected by failures such as shorted-bit and open-bit failures can be identified with a high degree of certainty.

Figure 6 is a graph as an illustrative example of the probability (p) that a particular cell will have a certain parametric value, in this case resistance (r), corresponding to a logical "0" in the left-hand curve, or a logical "1" in the right-hand curve. As an arbitrary scale, probability has been given between 0 and 1, whilst resistance is plotted between 0 and 100%. The resistance scale has been divided into five ranges. In range 601, the resistance value is very low and the predicted range represents a shorted-bit failure with a reasonable degree of certainty. Range 602 represents a low resistance value within expected boundaries, which in this example is determined as equivalent to a logical "0". Range 603 represents a medium resistance value where a logical value cannot be ascertained with any degree of certainty. Range 604 is a high resistance range representing a logical "1". Range 605 is a very high resistance value where an open-bit failure can be predicted with a high degree of certainty. The ranges shown in Figure 6 are purely for illustration, and many other possibilities are available depending upon the physical construction of the MRAM device 1, the manner in which the storage cells are accessed, and the parametric values obtained. The range or

ranges are suitably calibrated depending, for example, on environmental factors such as temperature, factors affecting a particular cell or cells and their position within the array, or the nature of the cells themselves and the type of access employed.

Referring again to Figure 5, step 504 comprises counting a number of physical failures, as identified in the comparison of step 503. Suitably, the count of parametric failures in step 504 is performed on the basis of the number of symbols 206 (each containing one or more bits) which are affected by the identified physical failures.

Step 505 comprises comparing the number of parametric failures, i.e. the number of failed symbols identified by parametric testing, against a predetermined threshold value. The number of physical failures can be represented in any suitable form. Depending upon the nature of the ECC scheme employed, some types of failure can be weighted differently to other types of failure. Since the data stored in the storage cells represents encoded data, it is expected that ECC decoding will not be able to recover the original data, where the number of parametric failures is greater than the maximum power of the ECC scheme. Hence, the threshold value is suitably selected to represent a value which is equal to or less than the maximum number of failures which the ECC scheme employed is able to correct. Preferably, the threshold value in step 505 is selected to be substantially less than the maximum power of the ECC decoding scheme, suitably of the order of 50% to 95% of the maximum power. In a particular preferred embodiment the threshold value in step 505 is selected to represent

about 50% to 75% and suitably about 60% of the maximum power of the employed ECC scheme. Preferably, the step 505 comprises determining the number of parametric failures to be greater than the threshold value, such that performing ECC decoding is expected (with a sufficiently high probability) not to be able to recover information from the encoded data. That is, where the number of parametric failures is greater than the threshold value, there is a greater than acceptable probability that information is unrecoverable from the encoded data.

Step 506 comprises determining whether or not to continue use of the set of cells corresponding to the accessed block of data, in view of the number of parametric failures which have been identified. If desired, remedial action can be taken as outlined in step 304.

The physical evaluation of Figure 5 is particularly useful as a test procedure immediately following manufacture of the device, or at installation, or at power up, or at any convenient time subsequently. In one example, the test procedure of Figure 5 is performed by writing a test set of data to the device and then reading from the device, or by any other suitable parametric testing. In particular, it is useful to apply the method of Figure 5 to identify areas of the MRAM device which are severely affected by systematic errors caused by manufacturing imperfections, and remedial action can then be taken before the device is put into active use storing variable user data. In the preferred embodiment, each sector comprises four codewords, and a sector is made redundant where any one of its four codewords contains a

number of parametric failures which is greater than the threshold value of step 505. A block of data such as an encoded sector 202 having a number of failed symbols greater than the threshold value is not used at all in the subsequent life span of the device, because the probability of unrecoverable data errors would be too high. The threshold value used in the test procedure is set such that at least one and preferably several failures occurring subsequently will be tolerated. In particular, the threshold value is set to allow further systematic failures to be tolerated together with at least one and preferably several random failures, in a block of data.

The parametric evaluation of Figure 5 is particularly useful in determining shorted-bit and/or open-bit failures in MRAM devices. A systematic failure, such as a half select or some forms of isolated bit failure, is not so easily detectable using parametric tests, but is more readily discovered by logical evaluation using ECC decoding as in Figure 4. Therefore, in particularly preferred embodiments of the present invention the logical evaluation of Figure 4 is combined with the parametric evaluation of Figure 5 to provide a practical device which is able to take advantage of the considerable benefits offered by the new MRAM technology whilst minimising the limitations of current available manufacturing techniques.

The MRAM device described herein is ideally suited for use in place of any prior solid-state storage device. In particular, the MRAM device is ideally suited both for use as a short-term storage device (e.g. cache memory) or a longer-term storage device (e.g. a solid-state hard disk). An MRAM device can be employed for both short term storage

and longer term storage within a single apparatus, such as a computing platform.

A magnetoresistive solid-state storage device and methods for controlling such a device have been described. Advantageously, the storage device is able to tolerate a relatively large number of errors, including both systematic failures and transient failures, whilst successfully remaining in operation with no loss of original data. Simpler and lower cost manufacturing techniques are employed and/or device yield and device density are increased. As manufacturing processes improve, overhead of the employed ECC scheme can be reduced. However, error correction coding and decoding allows blocks of data, e.g. sectors or codewords, to remain in use, where otherwise the whole block must be discarded if only one failure occurs. Therefore, the preferred embodiments of the present invention avoid large scale discarding of logical blocks and reduce or even eliminate completely the need for inefficient control methods such as large-scale data mapping management or physical sparing.